Synthesis of model features for fake news detection using large language models

Adam Wierzbicki[1,†], Andrii Shupta[2,*,†] and Olexander Barmak[2,†]

[1] *Polish-Japanese Academy of Information Technology, Koszykowa 86 str. 02-008, Warsaw, Poland*
[2] *Khmelnytskyi National University, 11, Instytuts'ka str., Khmelnytskyi, 29016, Ukraine*

Abstract

In recent years, the issue of fake news has become exceedingly pertinent due to its rapid dissemination through social media and online platforms. Detecting fake news requires the utilization of various methods and resources. This work proposes an approach to building a model for fake news detection using generative artificial intelligence and natural language processing. The main focus of the proposed approach is the synthesis of model features for fake news detection using generative artificial intelligence and Natural Language Processing. The use of this proposed approach not only facilitates the detection of fake news but also renders the detection process transparent and customizable by the user. Additionally, through the proposed method, users gain the ability to expand features and train the system to adapt to new types of fake news and their variations. Experimental results, presented both qualitatively through visual analytics and quantitatively through statistical indicators, convincingly demonstrate the effectiveness of the proposed approach in detecting fake news with satisfactory accuracy levels reaching 90% and provide users with sufficient interpretability of the obtained decisions. Overall, this research aims to create an approach for detecting fake news that may have a significant impact on addressing this issue in contemporary society.

Keywords

Fake news, Fake news detection, Natural Language Processing, LLM

1. Introduction

In modern society, the spread of fake news has become a serious challenge due to its rapid dissemination on social media and other online platforms. Dissemination of false or deceptive information can significantly influence individuals' perceptions and beliefs. Therefore, detecting and combating fake news has become an important task that requires the application of various methods and strategies to accurately separate true information from manipulative content.

CLW-2024: Computational Linguistics Workshop at 8th International Conference on Computational Linguistics and Intelligent Systems (CoLInS-2024), April 12–13, 2024, Lviv, Ukraine

* Corresponding author.

† These authors contributed equally.

adamw@pjwstk.edu.pl (A. Wierzbicki); andrii.shupta@gmail.com (A. Shupta); alexander.barmak@gmail.com (O. Barmak)

0000-0003-0075-7030 (A. Wierzbicki); 0009-0000-9771-5579 (A. Shupta); 0000-0003-0739-9678 (O. Barmak)

© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR-WS.org/Vol-3722/paper5.pdf

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

Social media has become the primary source of news, especially among the youth [1]. However, with the increasing popularity of this method of information consumption, the prevalence of misinformation, including false information and unsubstantiated claims, is also growing. Unfortunately, many of these platforms lack reliable mechanisms for verifying users and their publications, which contributes to the spread of false information. This misinformation may contain propaganda aimed against individuals, society, organizations, or political parties. Given the vast amount of content on social media platforms, detecting all instances of fake news becomes challenging, underscoring the relevance of automated machine learning classifiers.

Various social media platforms exist today where users can post and share news online. However, many of these platforms lack reliable mechanisms for verifying users and their publications, which contributes to the spread of false information [2]. This misinformation may contain propaganda aimed against individuals, society, organizations, or political parties. Given the vast amount of content on social media platforms, detecting all instances of fake news becomes challenging, underscoring the relevance of automated machine learning classifiers.

Typically, methods for detecting fake news are trained on data available at the time of training, and this data may become outdated in the future, hence new approaches such as the use of Large Linguistic Models (LLM) have emerged [3].

Building on previous experience [4], this research proposes an approach to synthesizing models for detecting fake news based on transparent and interpretable features obtained using generative artificial intelligence and Natural Language Processing from news texts. This approach allows the system to be adaptive to changes in fake news and to expand its capabilities for detecting new forms of misinformation. The research proposes an integrated approach that combines the effectiveness and power of Large Linguistic Models (LLM) with the transparency and interpretability of machine learning models. This allows for the synthesis of model features for detecting fake news using responsible and understandable AI principles based on ethics and transparency.

The main contributions of this work are:

- a method for synthesizing model features for detecting fake news using generative artificial intelligence and Natural Language Processing
- obtaining features for detecting fake news using the proposed method and successful validation of the obtained model, which demonstrated its ability to detect fake news qualitatively through visual analytics and quantitatively through statistical indicators (accuracy exceeding 90%)

The article is structured as follows. In the Related Work section, research related to fake news detection is reviewed, their advantages and disadvantages are analyzed, and the research objective is formulated. The Methods and Materials section presents the theoretical foundations of the proposed approach to feature synthesis using generative artificial intelligence and natural language processing. The Results and Discussion section presents the experimental results under the proposed approach and discusses them.

2. Related Work

This section discusses research dedicated to detecting fake news conducted using various methods and techniques. In the work [5], a new method for detecting fake news was presented, based on combining different features, including textual and user-based, and utilizing deep learning models. Algorithms such as convolutional neural network (CNN) extensions to graphs allow for the combination of dissimilar data types, resulting in an accuracy of 92.7%.

Another study [6] focuses on using linguistic features. They utilized a dataset consisting of two sets containing an equal number of true and fake news articles with a political theme. Text fields were used to extract linguistic and stylistic features and build bag of words TF and BOW TF-IDF vectors. Various machine learning models, including bagging and boosting methods, were then applied, achieving the highest level of accuracy.

In the study [7], two machine learning algorithms using character and word n-gram analysis to detect fake news were evaluated. Experimental results showed that character n-gram analysis combined with Term-Frequency-Inverted Document Frequency (TF-IDF) yielded better results, with accuracy reaching 96%.

The work [8] proposes a model based on theoretical principles for detecting fake news, examining news content at various levels, including lexicon, syntax, semantics, and discourse. Recognized theories in the fields of social and forensic psychology are used to represent news at each level and conduct fake news detection within the trained machine learning model. As an interdisciplinary study, their work aims to study potential patterns in fake news, improve interpretability in creating fake news features, and investigate relationships between fake news, deception/disinformation, and strategies aimed at increasing views.

Based on the analysis of related works, various shortcomings in approaches can be identified. One of them is the insufficient quality of the data on which the model is based. If the model is trained on incorrect or inadequate data, it may misclassify news. Another factor is the speed of news dissemination on the Internet. Fake news can quickly gain popularity and spread faster than any model can detect them. It is also important to consider that fake news may contain some truthful information, making their detection more challenging. Another reason is the evolution of technologies and approaches to creating fake news. Over time, new technologies emerge, allowing for the creation of more convincing fake news, and models created to detect previous versions of fake news may be ineffective.

Therefore, the main goal of this work is to develop and investigate an approach that allows for the detection and synthesis of features for machine learning models using artificial intelligence and natural language processing. The mentioned approach aims at the ability to detect fake news, train on new data, and improve accuracy of analysis. Considering the increasing importance of trust in the results of artificial intelligence algorithms, emphasis is placed on creating mechanisms that ensure the responsibility and effectiveness of analytical models.

3. Methods and materials

In order to move from subjective to objective evaluation of news for fakeness, it is necessary to propose an approach to the synthesis of models that would contain objective and such that can be calculated, indicators indicating the presence or absence of features of fake news. In addition, such an approach is necessary for the possibility of updating the model, since the mechanisms of creating fake news are constantly being improved.

Based on the fact that the essence of the proposed approach is the analysis of texts without a clear understanding of which features may indicate misinformation, it is proposed to use the capabilities of Large Linguistic Models (LLMs) in addition to empirical (linguistic, psychological, etc.) analysis. That is, a mechanism is needed that will allow the parameters from the analysis to be "determined". That is, by linguistic and logical analysis with the help of LLMs, including, identifying key factors that can influence the identification of fake news. The main efforts are focused on natural language searches, taking into account Internet resources, articles and examples, with the aim of understanding how they can display features of fake news.

In order to have such a tool, we used a new aspect - Prompt Engineering. Hint engineering is a groundbreaking approach that is revolutionizing the field of artificial intelligence and natural language processing. This is an important element that provides hints for managing Language Models (LMs) and Large Language Models (LLMs) such as GPT-X [9] and LLama [10]. Cue engineering provides specific input-output pairs for LLMs, increasing their efficiency, accuracy, and safety in various tasks.

Developers of LLMs (eg ChatGPT (OpenAI) [11]) provide helpful guidance on how to best use prompt engineering. After analyzing them, it is possible to develop application mechanisms for building prompts-requests for identifying the necessary features in the texts.

In AI speech generation systems, prompts gather information about the user's speech and create new speech samples that were not previously known. Query elements can include natural language, code, and multimodal prompts that consider images, videos, and other media. In our case, at the stage of synthesis of the model and its initial parameters, it is suggested that only the news text, or part of it, was used as an input element.

Therefore, in addition to identifying features of fake news by empirical means (linguistic, psychological, etc.), it is worth applying the capabilities of LLMs. Next, we will consider how it is proposed to be implemented.

3.1. Preliminary recommendations for the use of the mechanism of engineering prompts (Prompt Engineering) to detect features of fake news

Let's define a technique (a set of interconnected methods) for detecting features of fake news by means of hint engineering using LLMs.

Before writing prompts, you need to determine an effective approach for their creation [12]. Prompts should have several key characteristics to increase their effectiveness specifically for detecting features of fakeness. It is important to specify the so-called instructions before (at the beginning of) the request. It is also important to correctly define the desired response format. That is:

- Instructions should be placed at the beginning of the request. Special characters must be used to separate instruction and context sections.
- Instructions should be placed at the beginning of the request. Special characters must be used to separate instruction and context sections.
- Instructions should be specific, descriptive, and as detailed as possible regarding the desired context, outcome, scope, format, style, etc.

You need to determine the desired output format with the help of examples.

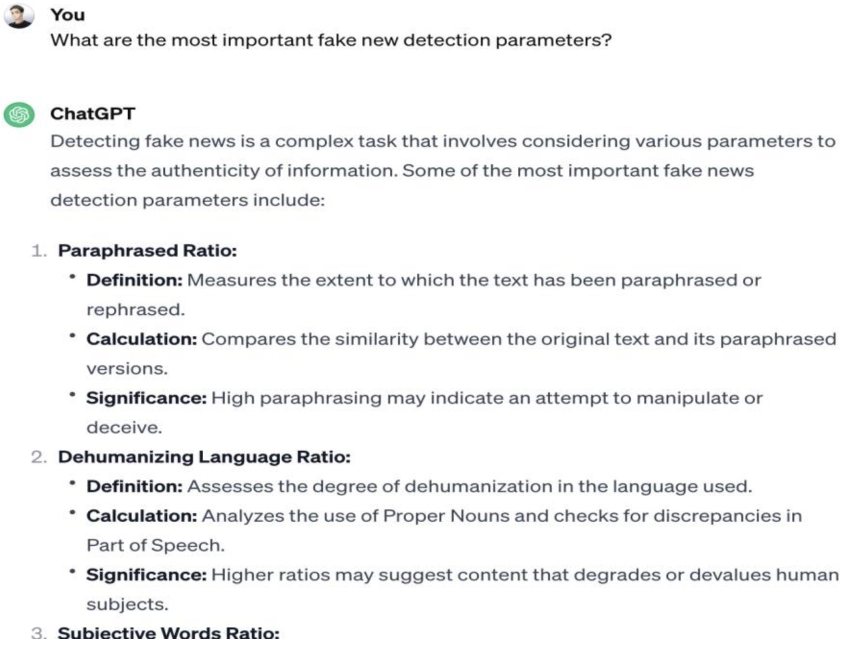
Next, in the dialogue mode with ChatGPT, using the experience gained from analyzing articles on the Internet, the main (often mentioned) criteria of fakeness are selected. Asking questions like:

1. what is most important in defining news?
2. what does this or that parameter mean?
3. how does it affect fakeness?

In the given way, you can get an initial set of basic criteria by which you can then try to create and test fakeness classifiers.

You can use both basic models (for example, gpt-3.5 [13]) and more advanced models (for example, gpt-4 model [14]), which will allow adding to the analysis, in addition to texts, images, audio-video files and fact-checking in real time.

In fig. 1 shows an example of the initial prompt for the problem under consideration.



You
What are the most important fake new detection parameters?

ChatGPT
Detecting fake news is a complex task that involves considering various parameters to assess the authenticity of information. Some of the most important fake news detection parameters include:

1. **Paraphrased Ratio:**
    - **Definition:** Measures the extent to which the text has been paraphrased or rephrased.
    - **Calculation:** Compares the similarity between the original text and its paraphrased versions.
    - **Significance:** High paraphrasing may indicate an attempt to manipulate or deceive.
2. **Dehumanizing Language Ratio:**
    - **Definition:** Assesses the degree of dehumanization in the language used.
    - **Calculation:** Analyzes the use of Proper Nouns and checks for discrepancies in Part of Speech.
    - **Significance:** Higher ratios may suggest content that degrades or devalues human subjects.
3. **Subjective Words Ratio:**

Figure 1: Initial prompt example

Next, for each found parameter, you need to understand the possibility of their determination in numerical form. It can be, for example, a fragment of program code using the methods of the corresponding NLP libraries, etc.

By analyzing different sources, you can get an understanding of what fake news often uses:

Paraphrasing to change the form of statements and avoid direct quotation. For example, in the article [15], the authors indicate that more than 70% of fake news contain paraphrasing elements.

Subjective words. Fake news tries to avoid explicit expression of the author's opinion and emotional judgments. Using ChatGPT, it is possible to emulate such an approach and determine how subjective words can affect the degree of trust in news. Phrases with subjectivity, such as "my own opinion," can serve as features of fake news.

Expressions that can cause a feeling of alienation or underestimation of human individuality. This may include using language that objectifies or creates the impression of hostility towards specific groups of people.

Headings that do not reflect the true content of the text. Using the LLM model, one can compare heading and content structures, paying attention to similarities.

Atypical or obscene speech can influence the definition of fake news. By analyzing atypical speech, it is possible to conduct a comparative analysis of fake news and real news and understand how it affects.

Tone of the text (emotional shades in the sentences). Fake news can use a tone aimed at creating negative emotions.

In addition to the above, it is also possible to aggregate various factors. It is possible to use the results of fact-checking to assess the reliability of the information provided.

To build a dynamic system, it is important to generate new parameters and use combinations of existing models. The user should be able to add parameters to the existing ones and form new prompts, which will be taken into account as additional weight coefficients for determining fake news.

So, the above are preliminary recommendations for using the proposed method of synthesizing model parameters to detect features inherent in fake news. Next, we present the main steps of the proposed method of model synthesis by means of generative artificial intelligence. Separately, we will repeat that it is possible to identify features empirically, using the approaches of linguistic and psychological analysis of texts. It is worth combining these approaches.

3.2. The main steps of the method

We will present the main steps of the proposed method of synthesis of model features by means of generative artificial intelligence.

Formally, you need to find the mapping:

$$T \rightarrow O, \tag{1}$$

where T is a set of news texts, O is a set of objective features that can be uniquely determined from the text and presented in numerical form immediately, or later, using the tools of NLP libraries.

The following is a sequence of steps for obtaining a display (1).

1. *Input information*. News or news that cause suspicion of fakeness.
2. The user asks a question in an arbitrary form, what is suspicious in the news.
3. The user receives an answer, possibly from several elements that can affect the fakeness of the news.
4. The user selects a parameter, and creates or adds to an existing prompt, for example, in the form:

Additionally, add THIS parameter to calculate fake news ratio from 0 to 1. The more of THIS we have in the text, the more we consider this news as a fake. Output explanation.

It is important to describe what this parameter does so that the system correctly forms and adds it.

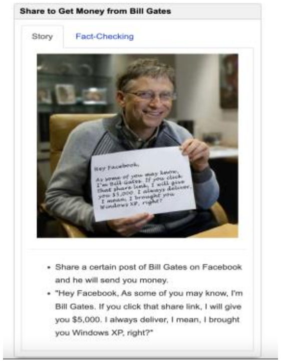
5. The system converts an arbitrary format to a new parameter of an existing input prompt:

```
THIS_parameter: this will detect how much it is a fake news
{
        "THIS_parameter_ratio_and_result",
}
```

The system gives the final result taking into account the free parameters.

6. *Output information*. The initial information of the method is:

- A prompt to a specific LLM that outputs the value of a feature in numeric form. Such a prompt can be used in the following methods to determine a specific feature vector element for a machine learning model.
- A prompt to a particular LLM that outputs the result verbally, but the researcher understands how it can be digitized using NLP libraries or otherwise.


Share to Get Money from Bill Gates
Story
Fact-Checking
Hey Facebook,
As some of you may know, I'm Bill Gates. If you click that share link, I will give you $5,000. I always deliver. I mean, I brought you Windows XP, right?

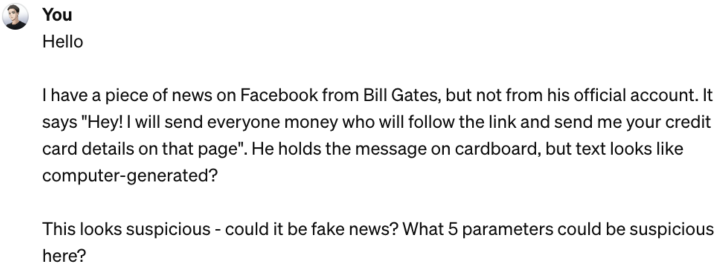
- Share a certain post of Bill Gates on Facebook and he will send you money.
- "Hey Facebook, As some of you may know, I'm Bill Gates. If you click that share link, I will give you $5,000. I always deliver, I mean, I brought you Windows XP, right?"

Figure 2: Example of fakes news

3.3. An example of using the method

We will give an example of using the method. The input information (original taken from article [16]) is: "Bill Gates will send money to everyone who clicks on the link he drops." It is obvious that this is a fake, and the picture and message are not real (picture 2 shows an example from a fake news site).

1. The user asks ChatGPT a question in an arbitrary form, about what is suspicious in the news (Fig. 3).



You

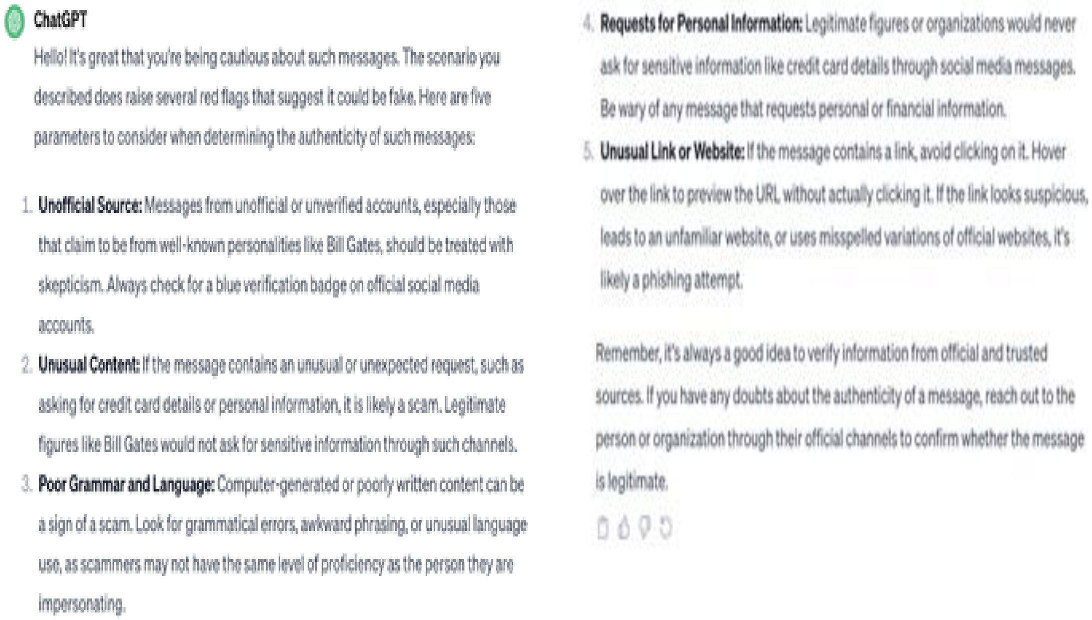
Hello

I have a piece of news on Facebook from Bill Gates, but not from his official account. It says "Hey! I will send everyone money who will follow the link and send me your credit card details on that page". He holds the message on cardboard, but text looks like computer-generated?

This looks suspicious - could it be fake news? What 5 parameters could be suspicious here?

Figure 3: Free form question

2. The user receives an answer from several elements that can affect the fakeness of the news (Fig. 4).



ChatGPT

Hello! It's great that you're being cautious about such messages. The scenario you described does raise several red flags that suggest it could be fake. Here are five parameters to consider when determining the authenticity of such messages:

1. **Unofficial Source:** Messages from unofficial or unverified accounts, especially those that claim to be from well-known personalities like Bill Gates, should be treated with skepticism. Always check for a blue verification badge on official social media accounts.
2. **Unusual Content:** If the message contains an unusual or unexpected request, such as asking for credit card details or personal information, it is likely a scam. Legitimate figures like Bill Gates would not ask for sensitive information through such channels.
3. **Poor Grammar and Language:** Computer-generated or poorly written content can be a sign of a scam. Look for grammatical errors, awkward phrasing, or unusual language use, as scammers may not have the same level of proficiency as the person they are impersonating.
4. **Requests for Personal Information:** Legitimate figures or organizations would never ask for sensitive information like credit card details through social media messages. Be wary of any message that requests personal or financial information.
5. **Unusual Link or Website:** If the message contains a link, avoid clicking on it. Hover over the link to preview the URL without actually clicking it. If the link looks suspicious, leads to an unfamiliar website, or uses misspelled variations of official websites, it's likely a phishing attempt.

Remember, it's always a good idea to verify information from official and trusted sources. If you have any doubts about the authenticity of a message, reach out to the person or organization through their official channels to confirm whether the message is legitimate.

Figure 4: ChatGPT answer

3. The user selects a parameter and adds a request to the system in the form:

"Additionally, add unusual_content parameter to calculate fake news ratio from 0 to 1. The more of unusual content we have in the text, the more we consider this news as a fake. Output explanation."

4. The system converts an arbitrary format to a new parameter of an existing input prompt:

```
"
        unusual_content_parameter: this will detect how much it is a fake news depending
if the content is expected for parties to work with.

        ...

        {
                "unusual_content_parameter_ratio_and_result",
}
"
```

5. The system gives the final result taking into account the free parameters:

"...The news is fake. Bill Gates would not ask for personal information for a reward... There was no mention of such messages, and he has 1 official account" etc.

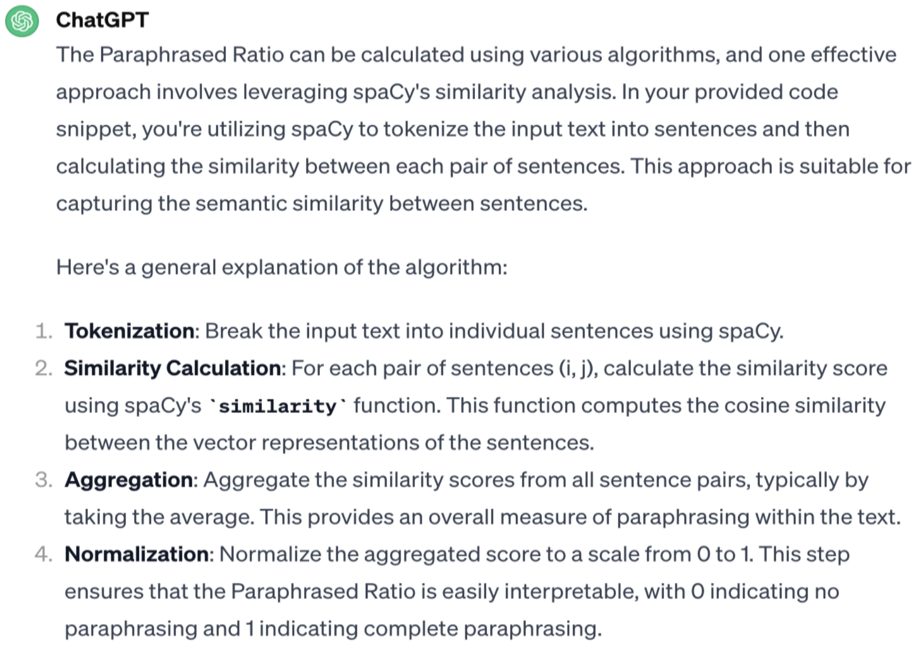
Therefore, the results obtained using the given method of synthesis of model parameters by means of generative artificial intelligence, together with the results obtained by the traditional method of identifying features (empirically, using the approaches of linguistic and psychological analysis of texts) will serve as input information for the method of forming a vector of features for machine learning models for detection of fake news.

4. Results and discussion

4.1. Synthesis of features

During interactive sessions with ChatGPT, six signature features were identified that show promising potential in identifying suspicious content, particularly in the area of fake news. These features were observed and refined during natural language interactions, which allowed us to form a preliminary set of criteria for the effective detection of deceptive or misleading information in text format. This section presents research findings, revealing the importance of each identified feature and their overall impact on improving text-based fake news detection capabilities. The following discussion follows from these findings and clarifies the significance of the features found in the broader plan of detecting fake news.

1. The paraphrase rate is a key parameter in our analysis, providing information about the degree of paraphrasing in a given text. During our studies of various texts, we found cases where certain information was presented in a way that deviated significantly from its original statement.
   Figure 5 shows the idea of the algorithm that was used to implement the code.



ChatGPT

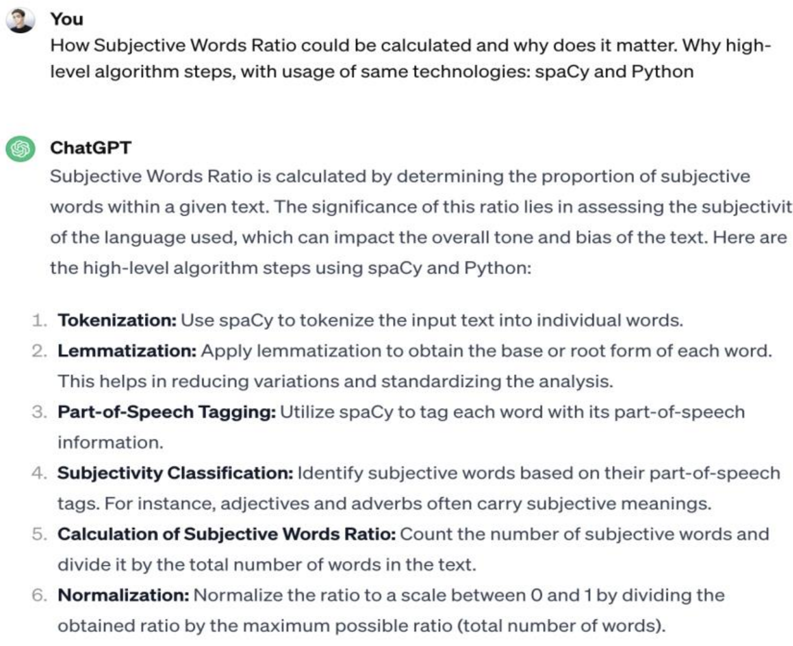
The Paraphrased Ratio can be calculated using various algorithms, and one effective approach involves leveraging spaCy's similarity analysis. In your provided code snippet, you're utilizing spaCy to tokenize the input text into sentences and then calculating the similarity between each pair of sentences. This approach is suitable for capturing the semantic similarity between sentences.

Here's a general explanation of the algorithm:

1. **Tokenization**: Break the input text into individual sentences using spaCy.
2. **Similarity Calculation**: For each pair of sentences (i, j), calculate the similarity score using spaCy's `similarity` function. This function computes the cosine similarity between the vector representations of the sentences.
3. **Aggregation**: Aggregate the similarity scores from all sentence pairs, typically by taking the average. This provides an overall measure of paraphrasing within the text.
4. **Normalization**: Normalize the aggregated score to a scale from 0 to 1. This step ensures that the Paraphrased Ratio is easily interpretable, with 0 indicating no paraphrasing and 1 indicating complete paraphrasing.

Figure 5: ChatGPT Paraphrase ratio algorithm

2. Subjective Words Ratio is calculated by determining the proportion of subjective words within a given text. The significance of this ratio lies in assessing the subjectivity of the language used, which can impact the overall tone and bias of the text.



You

How Subjective Words Ratio could be calculated and why does it matter. Why high-level algorithm steps, with usage of same technologies: spaCy and Python

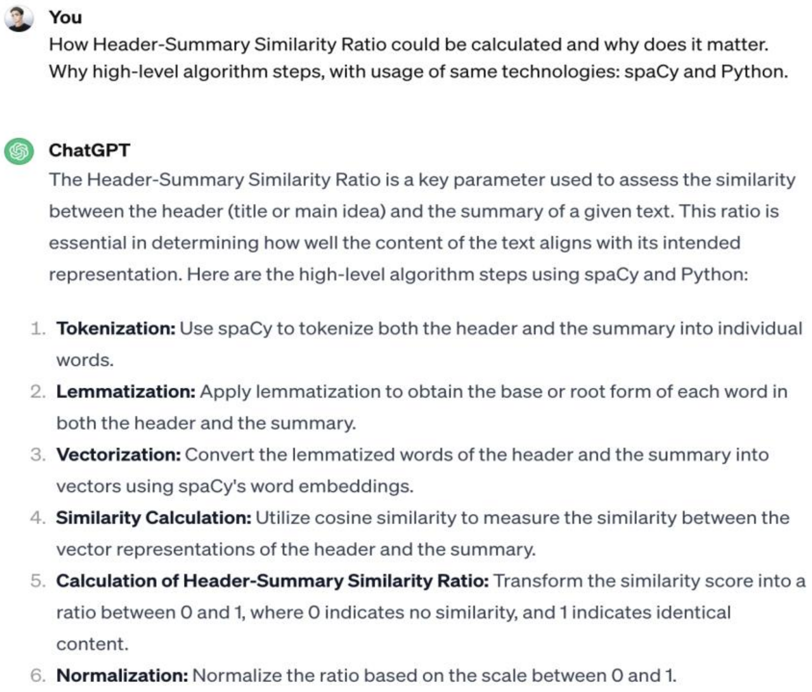
ChatGPT

Subjective Words Ratio is calculated by determining the proportion of subjective words within a given text. The significance of this ratio lies in assessing the subjectivity of the language used, which can impact the overall tone and bias of the text. Here are the high-level algorithm steps using spaCy and Python:

1. **Tokenization:** Use spaCy to tokenize the input text into individual words.
2. **Lemmatization:** Apply lemmatization to obtain the base or root form of each word. This helps in reducing variations and standardizing the analysis.
3. **Part-of-Speech Tagging:** Utilize spaCy to tag each word with its part-of-speech information.
4. **Subjectivity Classification:** Identify subjective words based on their part-of-speech tags. For instance, adjectives and adverbs often carry subjective meanings.
5. **Calculation of Subjective Words Ratio:** Count the number of subjective words and divide it by the total number of words in the text.
6. **Normalization:** Normalize the ratio to a scale between 0 and 1 by dividing the obtained ratio by the maximum possible ratio (total number of words).

Figure 6: ChatGPT Subjective Words Ratio algorithm

3. The Header-Summary Similarity Ratio is a metric used to evaluate the similarity between the header (title) and the summary (abstract) of a document. It measures how closely related the content of the summary is to the main topic or theme indicated by the header.



You

How Header-Summary Similarity Ratio could be calculated and why does it matter. Why high-level algorithm steps, with usage of same technologies: spaCy and Python.

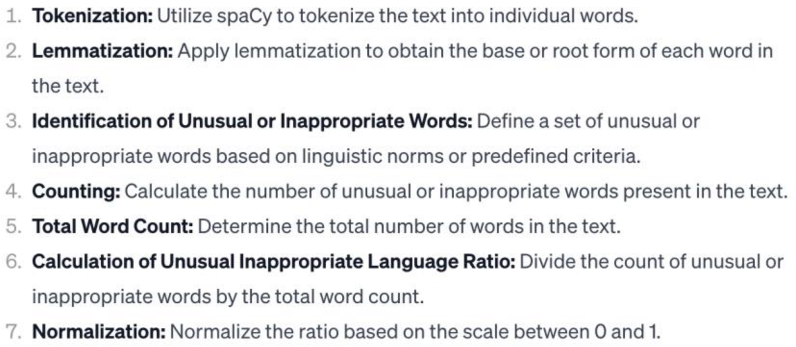
ChatGPT

The Header-Summary Similarity Ratio is a key parameter used to assess the similarity between the header (title or main idea) and the summary of a given text. This ratio is essential in determining how well the content of the text aligns with its intended representation. Here are the high-level algorithm steps using spaCy and Python:

1. **Tokenization:** Use spaCy to tokenize both the header and the summary into individual words.
2. **Lemmatization:** Apply lemmatization to obtain the base or root form of each word in both the header and the summary.
3. **Vectorization:** Convert the lemmatized words of the header and the summary into vectors using spaCy's word embeddings.
4. **Similarity Calculation:** Utilize cosine similarity to measure the similarity between the vector representations of the header and the summary.
5. **Calculation of Header-Summary Similarity Ratio:** Transform the similarity score into a ratio between 0 and 1, where 0 indicates no similarity, and 1 indicates identical content.
6. **Normalization:** Normalize the ratio based on the scale between 0 and 1.

Figure 7: ChatGPT Header-Summary Similarity Ratio algorithm

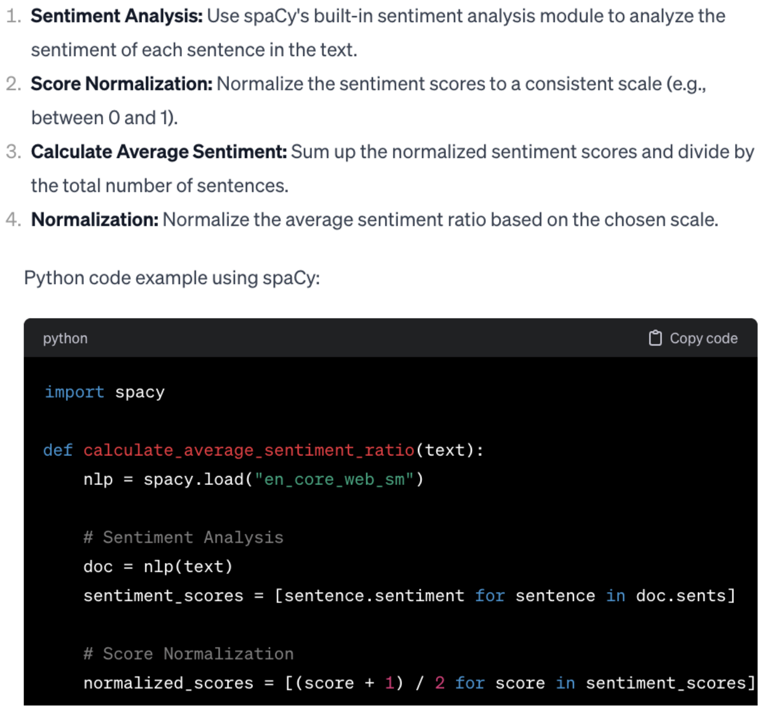
4. The Unusual Inappropriate Language Ratio is a parameter designed to quantify the presence of language that is either uncommon or inappropriate within a given text. Detecting such language is essential for assessing the overall tone and potential biases in the content



1. **Tokenization:** Utilize spaCy to tokenize the text into individual words.
2. **Lemmatization:** Apply lemmatization to obtain the base or root form of each word in the text.
3. **Identification of Unusual or Inappropriate Words:** Define a set of unusual or inappropriate words based on linguistic norms or predefined criteria.
4. **Counting:** Calculate the number of unusual or inappropriate words present in the text.
5. **Total Word Count:** Determine the total number of words in the text.
6. **Calculation of Unusual Inappropriate Language Ratio:** Divide the count of unusual or inappropriate words by the total word count.
7. **Normalization:** Normalize the ratio based on the scale between 0 and 1.

Figure 8: ChatGPT Unusual Inappropriate Language Ratio algorithm

5. The Average Sentiment Ratio is a parameter that measures the overall sentiment expressed in a given text. Sentiment analysis helps assess whether the text conveys positive, negative, or neutral emotions. Calculating this ratio is essential for understanding the emotional tone of the content.



1. **Sentiment Analysis:** Use spaCy's built-in sentiment analysis module to analyze the sentiment of each sentence in the text.
2. **Score Normalization:** Normalize the sentiment scores to a consistent scale (e.g., between 0 and 1).
3. **Calculate Average Sentiment:** Sum up the normalized sentiment scores and divide by the total number of sentences.
4. **Normalization:** Normalize the average sentiment ratio based on the chosen scale.

Python code example using spaCy:

```python
import spacy

def calculate_average_sentiment_ratio(text):
    nlp = spacy.load("en_core_web_sm")

    # Sentiment Analysis
    doc = nlp(text)
    sentiment_scores = [sentence.sentiment for sentence in doc.sents]

    # Score Normalization
    normalized_scores = [(score + 1) / 2 for score in sentiment_scores]
```

Figure 9: ChatGPT Calculating Sentiment with code example in Python

Including the Average Sentiment Ratio in our analysis is crucial for understanding the emotional context of the text. By considering sentiment, we gain insights into the overall tone, which can be valuable for evaluating the potential bias, subjectivity, and the persuasive nature of the content. This parameter contributes to a more comprehensive assessment of the text's characteristics in the context of fake news detection.

6. the next parameter Fact Checking Ratio we were able to determine only with the help of LLM, since LLM have access to the context and understand whether the facts really correspond to reality.

Without using LLM, we would need to use external resources or databases for validation. This step involves queries to reliable fact-checking databases or APIs to cross-check information. At this stage, we are satisfied with obtaining information using the LLM.

There were also intermediate and additional indicators that had a smaller impact on the results. Among them were such as dehumanizing language ratio, awkward text ratio and a separate definition of positive or negative sentiments. These parameters can be included to process, for example, tweets, where they can play a greater role in determining the "mood" of commenters.

A number of experiments were conducted to test the proposed approach and evaluate the validity of the feature vector. Below are their results and discussion. A description of the dataset used in the experiments is given. The result of the application of visual analytics to assess the ability of the proposed features of fake news texts to be divided into two classes is given. Visual and numerical (statistical metrics) results of classifier training (using SVM) are given. The discussion was carried out and the prospects of the proposed approach were given.

4.2. Dataset

The dataset [17] has over 20000 true and fake news labeled and categorized. It is very popular among the data science community and has been used in many articles and works.

4.3. Visual analysis with MDS

The results of applying the MDS method to input data (generalized features of news texts) in 2-dimensional space are shown in Figure 10.

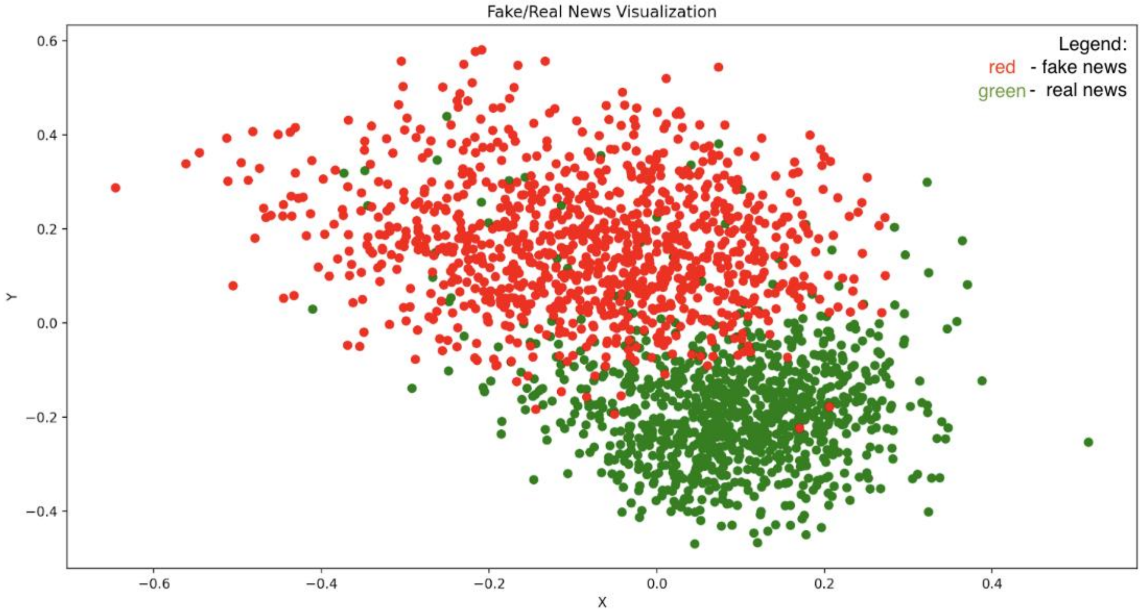


Figure 10: MDS Results for 2000 articles

As can be seen from Figure 10, the result is satisfactory, the classification was successful for a larger number of texts from the training set. Analysis of a small number of misclassified texts showed that there are true articles written with poorer text quality and vice versa.

4.4. Classification with SVM

After calculating the MDS, we can pass a value to the train_test_split method to split the data into training and test samples. Using SVM methods from the scikit-learn library, we obtained the following results:

Table 1
Comparison of the metrics for the classification problem

Number news, N	Precision	Recall	F_1
20	1.0	1.0	1.0
200	0.88	0.82	0.85
2000	0.93	0.92	0.93

After the number of news articles went over 2000, the results became consistent and we could consider it average for the whole dataset.

The obtained numerical results show the high accuracy of the proposed approach for determining fake news. The given values of the statistical metrics are either in the range or even better than the published modern results of other researchers.

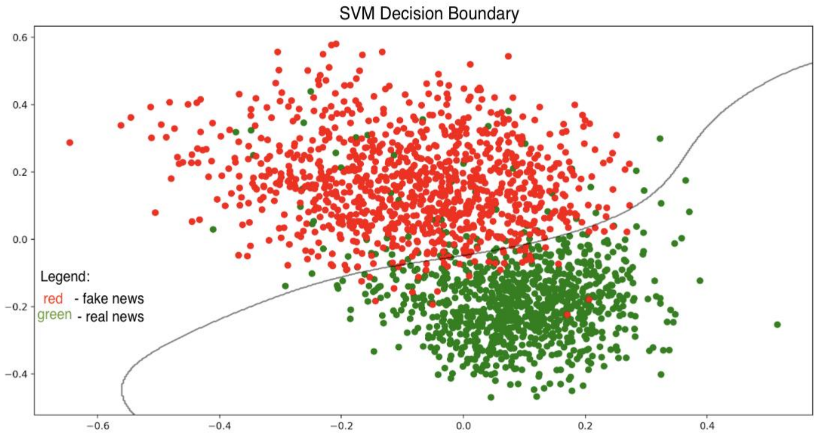


Figure 11: SVM decision boundary for 2000 elements

Also, in Figure 11 and 12, we can outline the decision boundary [18]. The boundary is determined by the support vectors, which are the data points closest to the hyperplane. The SVM then uses this decision boundary to classify new, unlabeled data points based on which side of the boundary they fall on.

4.5. Limitations

The main limitation of the proposed approach can be attributed to:

- lack of high-quality annotated training datasets (especially for the Ukrainian language) for successful classifier training and

- lack of generalized text features (to identify more hidden ways of creating fake news).

These limitations are not critical because the proposed model parameter synthesis method is capable of incorporating new datasets and new generalized features for retraining.

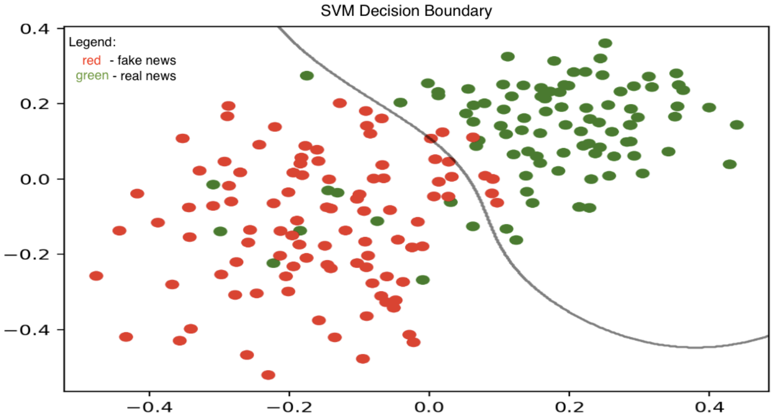


Figure 12: SVM decision boundary for 200 elements

5. Conclusion

The paper proposes an approach to the synthesis of features of models for detecting fake news using natural language processing techniques and machine learning algorithms. A thorough review of related works was conducted to ensure the novelty and effectiveness of the proposed approach. Six different parameters (common text features) were used for text modeling, and multidimensional scaling (MDS) was applied to obtain visual analytics as one of the criteria for evaluating the quality of the proposed approach. A support vector machine (SVM) classifier was trained to classify text into fake and non-fake news categories. The research results show that the proposed approach is in the same range or surpasses the existing methods in terms of accuracy (overall accuracy is more than 90%).

Limitations of the proposed approach include the lack of high-quality annotated datasets (especially for the Ukrainian language) to successfully train the classifier and the insufficiency of generalized text features (to detect more hidden ways of creating fake news). These limitations are not critical because the proposed model parameter synthesis method is capable of incorporating new datasets and new generalized features for retraining.

Future improvements of the approach will be aimed at increasing the accuracy of detecting fake news and achieving greater interpretability and understanding of the classification results.

References

[1] Shreya Ghosh, Prasenjit Mitra, Catching Lies in the Act: A Framework for Early Misinformation Detection on Social Media, 2023. doi: 10.1145/3603163.3609057.

[2] Qin Zhang a, Zhiwei Guo a, Yanyan Zhu b, Pandi Vijayakumar c, Aniello Castiglione d, Brij B. Gupta, A Deep Learning-based Fast Fake News Detection Model for Cyber-Physical Social Services, 2023. URL: https://www.sciencedirect.com/science/article/abs/pii/S0167865523000569?via%3Dihub

[3] Canyu Chen, Kai Shu, Can LLM-Generated Misinformation Be Detected?. Canyu Chen, Kai Shu. 2023. URL: https://arxiv.org/abs/2309.13788

[4] Krak I., Barmak O., Manziuk E., Kulias A. Data Classification Based on the Features Reduction and Piecewise Linear Separation, Advances in Intelligent Systems and Computing Volume 1072, Pp. 282 - 289, 2020, DOI: 10.1007/978-3-030-33585-4_28

[5] Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, Michael M. Bronstein: Fake News Detection on Social Media using Geometric Deep Learning, https://doi.org/10.48550/arXiv.1902.06673

[6] Mayank Kumar Jain; Dinesh Gopalani; Yogesh Kumar Meena; Rajesh Kumar: Machine Learning based Fake News Detection using linguistic features and word vector features, https://ieeexplore.ieee.org/document/9376576

[7] Hnin Ei Wynne, Zar Zar Wint: Content Based Fake News Detection Using N-Gram Models: https://dl.acm.org/doi/10.1145/3366030.3366116

[8] Xinyi Zhou, Atishay Jain , Vir V. Phoha , Reza Zafarani, Fake News Early Detection: A Theory-driven Model, https://dl.acm.org/doi/10.1145/3377478

[9] GPT-models list, URL: https://platform.openai.com/docs/models

[10] LLaMa LLM, Meta, URL: https://llama.meta.com

[11] Prompt engineering, Open AI, URL: https://platform.openai.com/docs/guides/prompt-engineering.

[12] Best practices for prompt engineering with the OpenAI API, URL: https://help.openai.com/en/articles/6654000-best-practices-for-prompt-engineering-with-openai-api

[13] GPT-3.5, URL: https://platform.openai.com/docs/models/gpt-3-5

[14] GPT-4, URL: https:/ /platform.openai.com/docs/models/gpt-4-and-gpt-4-turbo

[15] How to Identify Fake News, URL: https://www.kaspersky.com/resource-center/preemptive-safety/how-to-identify-fake-news.

[16] Fake News: Examples, URL: https://library-nd.libguides.com/fakenews/examples

[17] Clement Bisaillon, Fake and real news dataset, 2019, https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset

[18] SVM Decision Boundary, https://scikit-learn.org/0.18/auto_examples/svm/plot_iris.html